(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 766 428 A2

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
02.04.1997 Bulletin 1997/14

(51) Int. Cl.6: **H04L 12/40**, H04L 12/64

(21) Application number: 96115369.9

(22) Date of filing: 25.09.1996

(84) Designated Contracting States:
DE GB IT

(30) Priority: 26.09.1995 JP 270738/95

(71) Applicant: YAMAHA CORPORATION
Hamamatsu-shi, Shizuoka-ken 430 (JP)

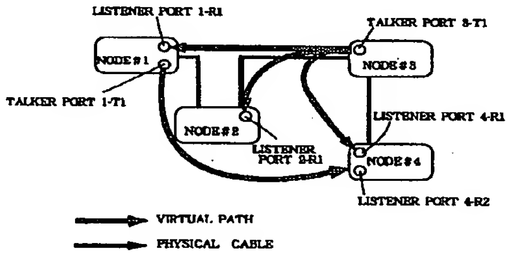
(72) Inventors:
- Fujimori, Junichi,
c/o Yamaha Corp.
Hamamatsu-shi, Shizuoka-ken 430 (JP)
- Abe, Tatsutoshi,
c/o Yamaha Corp.
Hamamatsu-shi, Shizuoka-ken 430 (JP)

(74) Representative: Kehl, Günther, Dipl.-Phys. et al
Patentanwälte
Hagemann & Kehl
Postfach 86 03 29
81630 München (DE)

(54) Local area network transferring data using isochronous and asynchronous channels

(57) In a network system, a plurality of nodes are communicable with one another to exchange information and to transfer data, and a bus is provided for connection to the nodes and for configuration of logical paths to logically link one node to another node so as to secure transfer of the data. Each node has at least one port which is allocated for accessing the bus and which is classified into four types of an isochronous talker, an isochronous listener, a multicast talker and a multicast listener. Each node binds an isochronous channel number and a communication band to the isochronous talker when the same is allocated for isochronously transmitting the data labeled by the bound isochronous channel number to the bus through the bound communication band. Each node binds an isochronous channel number to the isochronous listener when the same is allocated so that the isochronous listener can exclusively receive data transmitted from another isochronous talker allocated to another node if the transmitted data is labeled by the same isochronous channel number as that bound to the isochronous listener. Each node binds a multicast channel number to the multicast talker when the same is allocated for asynchronously broadcasting data labeled by the bound multicast channel number to the bus. Otherwise, each node binds a multicast channel number to the multicast listener when the same is allocated so that the multicast listener can exclusively receive data transmitted from another multicast talker allocated to another node if the transmitted data is labeled by the same multicast channel number as that bound to the multicast listener.

FIGURE 1



Printed by Rank Xerox (UK) Business Services
2.14.1/3.4

Description

BACKGROUND OF THE INVENTION

The present invention relates to a network system in which a multiple of nodes are interconnected to each other by logical paths established in a bus, and relates to a data transfer method which can preferably be employed in a network system for transmission of MIDI (Musical Instrument Digital Interface) data and audio data.

In a recent AV (Audio Visual) system, a multiple of devices can be connected to each other to form a network. The connection media such as coaxial cables, shield cables and parallel pair cables are used for connecting electronic devices to each other. The situation is the same as in connecting MIDI devices. In order to establish a network in MIDI systems, electronic musical instruments are interconnected to each other by the coaxial cables, shield cables and so on. As described above, in the current MIDI systems (or electronic music environments employing MIDI), inter-device paths are composed of a network topology structured by the physical connection media, so that it is troublesome to restore the same network configuration in another place.

Further in such a network, the connection cables tend to be so many lines that the cables occupy a large space, and re-cabling work thereof is painstaking once the cables are disconnected. Thus, there is proposed a network architecture in which multiple devices are connected to each other through a single cable, by which data are exchanged among the devices. The protocol layer structure in such a network architecture is shown in Figure 35. The protocol layer structure is comprised of a physical layer 102, a link layer 103, a transaction layer 104, and a serial bus manager 101. The physical layer 102 defines a physical interface interconnecting nodes, carries out conversion between an electric signal and logical symbols which can be handled by the link layer 103. The electric signal is transferred through various serial bus media. In the architecture, the single physical layer 102 can transfer data by certain bus arbitration. In other words, the multiple nodes never transmit data simultaneously. The link layer 103 takes in charge of addressing node to node, data check and data framing, to thereby provide one-way data transmission service for the transaction layer 104. The data transmission is acknowledged by an ACK signal from a data recipient node. The link layer 103 also provides an isochronous transmission described later. The transaction layer 104 provides inter-node transaction service (request-response protocol) recommended, for example, by IEEE 1212 CSR (Control and Status Register). However the transaction layer 104 does not provide any service for isochronous data. The serial bus manager 101 is an entity administrating the CSR which represents facilities of each node. The serial bus manager 101 carries out centralized administration of isochronous channels and frequency bands of the data within the local bus.

In the isochronous transmission, a bus cycle of 125 μsec is normally used so that a node having an isochronous channel can transmit one data packet within one cycle through a band allocated for the node. Further in the isochronous transmission, a packet is not addressed to a unique node, but a packet labeled by an isochronous channel number is broadcasted to all the nodes. In each node supporting the isochronous transmission, each reception of the isochronous packet is all notified to the link layer 103. The link layer 103 determines whether the transferred data packet should be accepted or not according to the channel number associated with the packet. Such a network architecture is proposed, for example, in IEEE P1394 "High Performance Serial Bus Standard". In the network architecture, the plural nodes are just required to connect with a single cable in the physical level. In the isochronous transmission, a transmitter node obtains a channel number and a band as resources for data transmission in advance, while a receiver node specifies a channel number to receive data from a corresponding transmitter node. Thus the isochronous packet broadcasted from a transmission port of the transmitter node can be received by a unique receiving port of the receiver node. As described above, the link layer 103 determines whether each transmitted isochronous packet should be accepted or not. The accepted packet is handed to an upper layer of the protocol layer structure. In such a network architecture where a certain transmission band must be secured in the isochronous transmission, transfer of discrete data such as a MIDI message may not be effective because the obtained band is not fully utilized, so that it is difficult to maximize the network usage efficiency.

SUMMARY OF THE INVENTION

A purpose of the present invention is to provide a network architecture and a data transmission method by which the maximum network usage efficiency can be realized even in case that the discrete data is transmitted.

Another purpose of the present invention is to provide a network architecture in which resetting or reconfiguration of the network can be automatically executed even in case that a node is newly installed or removed while the network is in operation.

According to the invention, a network system comprises a plurality of nodes communicable with one another to exchange information and to transfer data, and a bus provided for connection to the nodes and for configuration of logical paths to logically link one node to another node so as to secure transfer of the data. Each node has at least one port which is allocated for accessing the bus and which is classified into four types of an isochronous talker, an isochronous listener, a multicast talker and a multicast listener. Each node binds an isochronous channel number and a communication band to the isochronous talker when the same is allocated for isochronously transmitting the data labeled by the bound isochronous channel number to the bus through the bound communication band. Each node binds an isochronous channel number to the isochronous listener when the same is allocated so that the isochronous listener can exclusively receive data transmitted from another isochronous talker allocated to another node if the transmitted data is labeled by the same isochronous channel number as that bound to the isochronous listener. Each node binds a multicast channel number to the multicast talker when the same is allocated for asynchronously broadcasting data labeled by the bound multicast channel number to the bus. Otherwise, each node binds a multicast channel number to the multicast listener when the same is allocated so that the multicast listener can exclusively receive data transmitted from another multicast talker allocated to another node if the transmitted data is labeled by the same multicast channel number as that bound to the multicast listener.

In a specific form, a transmitter node having either of the isochronous talker and the multicast talker operates when the logical paths are reset for broadcasting talker information representative of resources including an isochronous channel number, a communication band and a multicast channel number, which may be newly bound, respectively, to the isochronous talker and the multicast talker upon resetting of the logical paths, while a receiver node having either of the isochronous listener and the multicast listener operates when the broadcasted talker information is received for newly binding resources including an isochronous channel number and a multicast channel number to the isochronous listener and the multicast listener to thereby restore the logical paths. Further, the receiver node saves path information representative of the logical paths and newly binds the resources according to the broadcasted talker information and the saved path information to ensure correspondence between the resources of the receiver node and the resources of the transmitter node in terms of the isochronous channel number and the multicast channel number.

In another specific form, each node has a communication architecture composed of protocol layers including a transport layer which contains an isochronous manager for acquiring isochronous resources and binding the acquired isochronous resources to either of the allocated isochronous talker and the isochronous listener, a multicast manager for acquiring multi-cast resources and binding the acquired multicast resources to either of the allocated multicast talker and the multicast listener, and a path information manager cooperating with the isochronous manager and the multicast manager for providing an upper protocol layer with a service to set the logical paths and to manage the set logical paths.

In a further specific form, a transmitter node is allocated with both of the isochronous talker and the multicast talker such that the transmitter node repeatedly carries out a transfer cycle containing isochronous transmission of data by the isochronous talker and asynchronous broadcasting of data by the multicast talker, and such that the transmitter node distributes the data each transfer cycle to either of the isochronous talker and the multicast talker according to property of the data and availability of the isochronous talker and the multicast talker. For example, the transmitter node treats mixture of continuous music data and discrete music data, and distributes the continuous music data to the isochronous talker while distributes the discrete music data to the multicast talker.

As set forth in the foregoing, according to the present invention, virtual logical paths can be configured within the network system independently from the physical connection topology. If two nodes or devices are connected to the network system, a logical path can be set between the two devices independently from the physical location of the devices, and data can be exchanged through the logical path. Thus, the data can be exchanged via the virtual logical path which does not depend on the network topology, so that the connecting sequence of the devices is never mistaken, and the data can be transferred reliably. The logical connection can be modified easily without changing the physical connection of the devices, since the virtual path is configured logically among the devices. The virtual path information representative of the logical path configuration is stored in each device connected to the network. Therefore, all the path information of the whole network system can be saved in a memory media, and the same network system configuration can be restored easily and instantly with loading the saved path information from the memory media. Further, the present invention makes it possible to use not only the isochronous transmission in which the band of the transmission is assured, but also the multicast transmission with which data can be transmitted on demand to specified ones of the multiple nodes. Thus, the data can be transferred efficiently no matter how the discrete data is generated. Further, the music data such as the MIDI data and the audio data, which requires real time transmission, can be transmitted efficiently.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a schematic block diagram illustrating data communication in an mLAN according to the present invention.

Figure 2 is a schematic block diagram illustrating protocol layer structure in the mLAN according to the present invention.

Figure 3 is a schematic block diagram illustrating multicast transmission in the mLAN.

Figure 4 is a table diagram showing a format of a path information table.

Figure 5 is a table diagram showing a format of a node information table.

Figure 6 is a table diagram showing a format of a multicast information table.

chronous transmission occupies the band continuously, so that the isochronous transmission does not function efficiently for transmitting discrete data such as MIDI messages. Thus, in addition to the isochronous transmission, there is introduced in the mLAN an asynchronous multicast transmission, by which data can be transmitted from one node to a multiple of other nodes. The data can be transmitted from one node to other nodes by both of the isochronous transmission and the multicast transmission. A "channel" concept is also introduced for the asynchronous packet transmission in the mLAN.

The introduced asynchronous multicast transmission is a sort of "listener initiative" in which a listener (recipient) selectively receives a packet broadcasted by a talker (transmitter). In this method, a single band can be utilized efficiently since a common packet may be well transmitted if the same message should be sent to multiple destinations. Further in this method, the packet selection is under the control of the listener, so that work load of the talker never changes even when the number of the receiving nodes is increased. Thus, devolving a large part of the data transfer control to listeners, the work load of talkers can be decreased, and real time performance of the data transmission can be improved.

1.2 mLAN protocol layer structure

Figure 2 shows an example of the mLAN protocol layer structure. Figure 2 illustrates the protocol layer structure of the inventive mLAN corresponding to OSI (Open System Interconnection) reference model. The protocol layer structure can be divided into a lower layer infrastructure and an upper layer infrastructure. The lower layer infrastructure corresponds to the first to fourth layers of the OSI reference model, while the upper layer infrastructure corresponds to the fifth to seventh layers of the OSI reference model. Means to exchange messages between applications is specified in the lower layer infrastructure, while contents of the message are specified in the upper layer infrastructure. The lower layer infrastructure establishes logical paths and provides end-to-end transmission service for the upper layers. The upper layer infrastructure defines semantics of the messages and specifies application services to utilize the messages.

The upper layer is comprised of mLAN protocols 11 corresponding to the presentation layer (OSI sixth layer) 26, and an mLAN application 12 corresponding to the application layer (OSI seventh layer) 27. The functions implemented in the session layer (OSI fifth layer) 25 are carried out by the lower layers including the mLAN transport layer 10, so that the mLAN upper layer infrastructure has no layer corresponding to the session layer 25. The group of the mLAN protocols 11 specifies various supplemental control mechanisms to enable effective transmission of musical information, and provides various data formats and communication protocols. The protocols include a MIDI transfer protocol, a Digital Audio transfer protocol and so on. The mLAN application 12 utilizes a data exchange method defined by the mLAN protocols 11.

The lower layer infrastructure is comprised of an mLAN transport layer 10 corresponding to the transport layer (OSI fourth layer) 24, a transaction layer 3 corresponding to the network layer (OSI third layer) 23, a link layer 2 corresponding to the data link layer (OSI second layer) 22, a physical layer 1 corresponding to the physical layer (OSI first layer) 21, and a node controller 4.

In the mLAN transport layer 10, a sub-address called "port" is defined to achieve a port-to-port data communication, and to absorb difference of the specification in the lower layer. The mLAN transport layer 10 also provides a common service interface independent on the specification of the lower layer for the mLAN upper layer. Thus, if there are lower layers structured based on different standards, the mLAN transport layer 10 is provided for each lower layer. In the mLAN transport layer 10, there are defined a PIM (Path Information Manager) 8, an isochronous manager 7, a multicast manager 6, a transaction manager 5 and an NIM (Node Information Manager) 9.

In the physical layer 1, a physical interface interconnecting multiple nodes is specified, and the physical layer 1 carries out conversion between an electric signal and logical symbols which can be handled by the link layer 2. In the mLAN, normally a cable is used for physical connection.

The link layer 2 takes in charge of addressing node to node, data check or examination, and data framing so as to provide one-way data transfer service for the transaction layer 3. The data transmission is acknowledged by an ACK signal from the data recipient. The link layer 2 also provides an isochronous transmission described later.

The transaction layer 3 provides inter-node transaction service (request-response protocol) recommended, for example, by IEEE 1212 CSR (Control and Status Register) architecture. The node controller (which corresponds to the serial bus manager) 4 is an entity administrating the CSR which represents facilities of each node.

1.3 mLAN transport layer

In the mLAN transport layer 10, the sub-address called "port" is defined to achieve port-to-port data communication. The mLAN transport layer 10 supports not only the isochronous transmission and the asynchronous packet transaction, but also the multicast transmission which is a data transfer protocol to send data from one port to a multiple of other ports.

1.4 Addressing in data transmission

In the inter-port data transfer by the multicast transmission and the isochronous transmission, a channel should be bound to a port prior to the actual transmission. The transmitter node should obtain a channel number as a resource to transfer data in advance, while the receiver node specifies a channel number to receive data from a talker port of the transmitter node. The channel number is one of the resources allocated dynamically to transmitting and receiving ports in the data transfer. A different channel number may be allocated for the port before and after bus reset by which the paths are initialized. The receiver node specifies the corresponding transmitter node by means of a node ID. A unique node ID is automatically allocated to each node upon the bus reset. The user need not configure the node ID in contrast to SCSI (Small Computer System Interface) ID allocation. The uniqueness of the node ID in the local bus is guaranteed by the dynamic allocation by the physical layer. Thus, the node ID and the channel number used for addressing are re-allocated upon the bus reset. The node ID and the channel number may be different before and after the re-allocation. Therefore, data transfer may not be resumed due to change of the address mapping, if a bus reset occurs in the middle of the isochronous/multicast transmission. Considering such situation, route information called "path" is saved in the mLAN transport layer 10 to restore the data transfer route, so that the data transfer can be resumed even when the node ID and the channel number are changed upon the bus reset. The user does not care for the bus reset.

1.4.1 Path

The path set in the port which executes the isochronous/multicast transmission is not connection-oriented, but is a kind of route information to transfer data to multiple receiving ports. In this case, the data is transferred in connectionless manner. The path information is saved in the destination node of the route and the path is restored by the mLAN transport layer 10 after the bus is initialized by bus reset or power-on reset according to the saved path information. The configured paths are administrated by the PIM (Path Information Manager) 8.

In the isochronous/multicast transmission, a packet is transferred from a talker port of one source node to listener ports of plural destination nodes. It would be difficult to transfer data in real time if the source node resolved the routing due to increase of the computing load of the source node. For example, if the source node has a list of destinations and a listed destination node is not found in the network, the source node must resolve the routing. Further, there may be simple source nodes such as a mouse and a keyboard for instance, which cannot allocate memory enough to store path information for a number of destination nodes. Thus, the path information is stored in the destination nodes according to the invention.

As described above, the path information is saved in the receiver nodes in order to reconstruct the routing after the node ID and the channel number is changed by bus reset. Thus, the path information should be a type of information which is never affected by the bus reset The path information may include:

Talker port ID
Talker Node Unique ID
Listener port ID

as illustrated in the table of Figure 4.

The port ID is determined by applications, and Node Unique ID is hard-coded in a device of the node at the shipping of the device product. The Node Unique ID is never changed by the bus reset. Upon bus reset, the mLAN transport layer 10 updates internal path information depending on changes in the channel number and the node ID in order to provide the routing resource independent on the bus reset to the upper layer applications. Thus, applications can resume data transfer regardless of the bus reset.

1.4.2 PIM (Path Information Manager)

The PIM (Path Information Manager) 8 defined in the mLAN transport layer 10 is a module administrating each of the paths set between a pair of ports. The PIM 8 handles the allocating and releasing of the path, and maintains the path information. The PIM 8 also has a function to restore the logical paths according to the path information stored in the node, automatically upon bus initialization procedure caused by addition or removal of a node device, or upon power-on reset. The PIM 8 provides a service to set the logical paths among ports which execute isochronous or multicast transmission for the upper layer applications. The path information is stored in the PIM 8 as a path information table shown in Figure 4. The addition or removal of a physical node device causes a bus reset. On the bus reset, node ID is dynamically allocated so that the node ID may be changed before and after the bus reset. Of course, this node ID change affects node addressing. The data could not be transferred correctly without reconfiguration. Thus, upon such a bus configuration change, the PIM 8 updates the path information table in order to assure the correct data transfer. Thus, the upper layer applications need not care for the change in the bus configuration. Preferably, the path information stored in the PIM 8 is retained while the power is turned off, and thus the correct paths can be restored after power-on reset.

2. mLAN transport layer specification

2.1 Services in mLAN transport layer

The mLAN transport layer 10 provides three kinds of transmission services for the upper layer. The three services include the multicast transmission, isochronous transmission, and transaction, which are characterized as described below.

A. Multicast transmission
The multicast transmission is an asynchronous data packet transfer in which a data packet is transferred from a talker port to multiple listener ports. The multicast transmission is specified by the mLAN transport layer 10. In the multicast transmission, packets are broadcasted to destinations, and an ACK is never returned. Thus, the transmitter node cannot recognize whether the data transfer is successful or not, but frequency bands of the bus can be utilized efficiently.
B. Isochronous Transmission
The isochronous transmission is specified in the link layer 2. Each node secures a band before the same commences the data transmission to assure a certain signal delay range, so that the total load of the whole network system is fully controlled. This transmission method is suitable for situations in which a constant volume of data is polled periodically, or in which a data arrival timing must be accurately controlled.
C. Transaction
The transaction is a bidirectional peer-to-peer transmission method specified in the transaction layer 3. The transmission is executed in connectionless manner, but ACK/retry function is implemented to assure reliable data transfer.

2.1.1 Port

The port is a service access point in the mLAN transport layer 10, and will be explained hereunder. The upper layer can access the service provided by the mLAN transport layer 10 through the port. The above-described transmission services are available only between the ports having particular property. In other words, a packet cannot be distributed to a port which is improper for the relevant service. The property of the port is defined by the resource bound thereto. The resources which can be bound as the port property can be categorized as listed below.

A. Transaction Port
A port handling transaction. The transaction port carries out both of transmitting and receiving of data. An address within the node is bound as the resource.
B. Isochronous talker
A port handling the isochronous data transmission. An isochronous channel number and an isochronous bandwidth are bound as the resources.
C. Isochronous listener
A port handling the isochronous data reception. An isochronous channel number is bound as the resources. The listener can receive a packet labeled by the same channel number as that bound to the listener.
D. Multicast talker
A port handling multicast data transmission. A multicast channel number is bound as the resources.
E. Multicast listener
A port handling multicast data receiving. A multicast channel number is bound as the resources. The listener can receive a packet labeled by the same channel number as that bound to the listener.

The transaction in the mLAN transport layer 10 can be recognized by the level of the transaction layer 3 as a transaction to the port-bound address. In this method, it is impossible to access an address location (an entry in configuration ROM, for example) which is not bound to any port. In order to access such a resource, the NIM (Node Information Manager) 9 is provided according to the invention. By issuing a request to the NIM 9 of a certain node, other nodes can derive node information (the Node Unique ID, number of ports in the node, properties of the ports) of the certain node.

2.2 Structure of mLAN transport layer

As described above, the mLAN transport layer 10 is comprised of five modules listed below.

talker information update procedure is executed upon receiving the talker information reporting packet, which is broadcasted when a talker port is created in a node. The talker information is updated depending not only on the talker information reporting packet broadcasted from the own node, but also on the talker information reporting packet broadcasted from the other nodes.

2.3.11 Communicating with NIM

For the isochronous manager 7, the NIM issues a service primitive "Isochronous Manager Control Request", which includes parameters listed below.

RESET:	Resets the status of the isochronous manager 7.
INITIALIZE:	Initializes the status of the isochronous manager 7.
TALKER INFO RECEIVED:	Notifies that the NIM 9 has received a talker information reporting packet.

A service primitive "Isochronous Manager Control Confirmation" is issued from the isochronous manager 7 to NIM 9 in order to return the results of the control request made by the "Isochronous Manager Control Request". The status of the isochronous manager 7 is reported by a service primitive "Isochronous Manager Event Indication" issued from the isochronous manager 7 to the NIM 9.

2.4 Multicast transfer service

The multicast transfer service is described below. The multicast data transfer is an asynchronous and connectionless data communication, which is performed from a transmitting port to multiple receiving ports. In the multicast data transfer, the transmitting port is called a "multicast talker", and the receiving port is called a "multicast listener". The multicast data transfer is defined in the mLAN transport layer 10. The application user should bind a multicast channel number to a talker port before starting transmission via the port. This means that the talker port acquires a right to send data through the channel. Only the multicast talker can actually transmit a multicast packet using a certain multicast channel provided that the multicast talker is bound with the channel number. The packet sent from the talker is not addressed to a unique or individual multicast listener, but the packet is broadcasted to all the nodes by the broadcast facility defined in the transaction layer 3.

If a multicast packet arrives at a node, the transaction layer 3 of that node notifies the multicast manager 6 of the arrival. The received packet is attached with a multicast channel number, which is allocated to the corresponding multicast talker, so that the multicast manager 6 of the designation node determines whether the packet should be accepted or not by examining the attached channel number. Thus, the multicast talker cannot recognize which of the multicast listeners is actually listening.

The multicast channel number bound to the multicast talker is acquired by the binding operation of the multicast manager 6 in the mLAN transport layer 10. The multicast channel number bound to the multicast listener is selected from ones which are already bound to the multicast talkers.

The mechanism of the multicast transfer is illustrated in Figure 3. In Figure 3, ports #1 and #3 of a node #1 functions as a multicast talker. These ports are respectively bound with multicast channel numbers #2 and #4. If a host application issues a transmission request for the port #1 of the node #1, the data is broadcasted to all the nodes in the local bus using the channel #2. Each node receives the transmitted packet by the broadcast facility. Upon receiving, the received packet is passed to the mLAN transport layer 10 from the lower layer which picks up the packet. In the nodes #2 and #3, either of the ports #2 and #3 is functioning as a multicast listener. The mLAN transport layer 10 in each node passes the data of the packet transmitted through multicast channel #2 to its own multicast listener port. Thus, the multicast transfer is carried out. Similarly, the packet delivered to the port #3 of the node #1 is transferred through the channel #4 to each node. The packet is received by the port #3 (bound with the multicast channel number #4) of the node #2. On the other hand in the node #3, the multicast channel #4 is not bound to any port, so that the mLAN transport layer 10 of the node #3 abandons the packet received through the channel.

2.4.1 Multicast manager

The multicast manager 6 is a module to obtain a multicast channel number for a port to execute the multicast transfer, and to bind the same to the port. The multicast manager 6 obtains a multicast channel number by issuing a talker information reporting packet. Further, the manager 6 receives all the talker information reporting packets issued from other nodes within the local bus to create a multicast information table containing correspondence between the multicast channel numbers and the multicast talker ports as shown in Figure 6. Upon receiving the talker information reporting packet, the multicast manager 6 of each node determines the talker's multicast channel number depending on the order of the reception while organizing the multicast information table shown in Figure 6. If the multicast manager 6 receives a talker information reporting packet transmitted from its own node, the multicast manager 6 determines the multicast channel number of its own node. The multicast information table is accessed by a multicast listener to search a multicast channel number bound to a corresponding multicast talker. The multicast manager 6 provides the binding information upon a request from the host application. The multicast manager 6 is provided in every node which has a multicast port (multicast talker/multicast listener) to execute the multicast data transfer.

2.4.2 Obtaining and binding multicast resources

The procedure for obtaining and binding multicast resources is described below referring to Figure 17. The multicast manager 6 executes the procedure for obtaining and binding multicast resources for a port to create a multicast talker. The procedure is commenced when the multicast manager 6 receives a "Multicast Manager Bind Resource Request" from the upper layer PIM 8. In step S600, the multicast manager 6 broadcasts a talker information reporting packet for all the nodes. The talker information reporting packet is formatted as shown in Figure 9, and contains an obtained multicast channel number associated with the Node Unique ID and the port ID. If it is recognized that a multicast channel number is successfully obtained in step S610, the obtained resource (multicast channel number) is bound to a specified multicast talker in step S620. Then, the PIM is notified of the result of the resource binding. In this case, the PIM 8 is notified that the binding is successful. On the other hand, if it is recognized that the multicast channel number allocation is failed, the procedure goes forward from step S610 to S630, and the corresponding notification is passed to the PIM 8, which terminates the procedure. The procedure of steps S610 and S620 is shown in dotted blocks, because the actual resource binding process is done by broadcasting a talker information reporting packet in the talker information update procedure described below.

2.4.3 Talker information update

The NIM 9 in each node receives the broadcasted talker information reporting packet, by which the talker information update procedure is controlled. The talker information update procedure is explained hereunder referring to Figure 21. The procedure is launched when the NIM 9 notifies the multicast manager 6 of the reception of the talker information reporting packet. In step S750, in order to allocate a multicast channel number to each node in the order of arrival of the talker information reporting packet, a packet counter provided in the multicast manager 6 in each node is incremented by "1". The multicast manager 6 has the packet counter, which is reset to "0" upon the bus reset and which counts received talker information reporting packets after the bus reset. In step S760, the multicast manager 6 updates the multicast information table (Figure 6) containing the correspondence among the Node Unique ID, port ID, and multicast channel number according to the received talker information reporting packet. The port ID is extracted from the talker information reporting packet, and the value of the packet counter is used as the resource information in order to create a talker entry. Then in step S770, the multicast manager 6 sends the talker information reporting packet to notify the PIM 8 of the update of the path information. In step S780, the NIM 9 is notified of the result of the update, which terminates the procedure. The multicast manager 6 which broadcasted a talker information reporting packet uses the counter value at the time of receiving the own packet as the multicast channel number to be bound to a multicast talker of the own node. The maximum number of multicast channels in the local bus is set to 256 (0 to 255). If the packet counter value exceeds the maximum number, the multicast manager 6 reports that fact to the upper layer.

2.4.4 Multicast data transmission

The multicast data transmission procedure is described below referring to Figure 19. The multicast data transmission is initiated when the multicast manager 6 receives data to be transferred from the upper layer PIM 8. In step S670, the multicast manager 6 which has received the data to be sent searches the multicast information table shown in Figure 6 in order to obtain the multicast channel number bound to the multicast talker to be used. Then, step S680 issues a multicast transfer request while sending the data to be transferred to the transaction layer 3. Thus, the data is multicasted. However in this method, the data is actually broadcasted so that ACK is never returned.

2.4.5 Binding multicast resources (Listener)

In order to receive a multicast packet by a multicast listener, the same multicast channel number contained in the multicast packet should be provisionally bound to the recipient listener. The binding of the multicast resources is illustrated in Figure 18. The binding procedure is commenced upon receiving a "Multicast Manager Bind Resource Request" from the PIM 8. First in step S640, the multicast information table is searched, and the specified port is bound with the multicast channel number by which the data is identified. In step S650, the port ID, port type, and multicast resource (multicast channel number) are written as a new entry into the multicast information table. In step S660, the PIM 8 is notified of the result of the binding procedure. The listener resource binding never fails, because there is only a binding request to bind the resource to the channel which has already been allocated to a corresponding talker. Thus, in this case, the PIM 8 is notified of the successful binding, thereby terminating the procedure. After the multicast resource is bound to the muiticast iistener, the muiticast data can be received correctly.

2.4.6 Multicast data reception

The multicast data reception procedure is explained hereunder referring to Figure 20. The procedure starts by the notification of the data reception from the transaction layer 3. First of all in step S700, the talker duplication is detected if any. The multicast channel number attached to the multicast packet is that bound to the multicast talker which has sent out the packet. Only one multicast talker deserves to be bound with a unique multicast channel number. In other words, each multicast talker transmits packets using a particular multicast channel number. Thus, if the multicast manager 6 recognizes that two or more multicast talkers are sending packets concurrently with the same multicast channel number, the procedure jumps forward to step S730, where the notification of data arrival to the PIM 8 is stopped, and the duplication is reported to the PIM 8 with sending the duplicated channel number information. On the other hand in step S700, if the talker duplication is not detected, the multicast manager 6 searches the multicast information table to identify the attending listener port. Then in step S720, if the received packet has the channel number that coincides with the channel number bound to the multicast listener in the node, the packet reception is notified to that listener. The multicast manager 6 picks up all the multicast packets upon the notification of data arrival from the transaction layer 3. However, the packet arrival is notified to the multicast listener port in the node, only when the multicast channel number coincides with that bound to the listener. Thus, the notified listener can exclusively receive the multicast data packet. If there are two or more destination listeners, the multicast manager duplicates the data to deliver the data to these listeners. In case that the attached multicast channel number is not bound to any listener in the node, the packet is abandoned. As described above, the multicast data reception procedure is executed correctly.

2.4.7 Communicating with lower layer

As shown in the foregoing, the multicast manager 6 in the mLAN transport layer 10 communicates with the transaction layer 3 in order to make the multicast transmission. The mLAN transport layer 10 uses service primitives listed below, which are defined in the transaction layer 3.

Transaction Data Request: the multicast manager 6 issues this primitive to the transaction layer 3 in order to initiate the multicast transmission. The multicast transmission is carried out using the broadcast facility in the transaction layer 3, so that a property BROADCAST WRITE is specified as a Transaction Type.

Transaction Data Indication: with this primitive, the multicast manager 6 is notified of the data arrival by the transaction layer 3.

2.4.8 Communicating with PIM

The multicast manager 6 communicates also with the PIM 8. The service primitives listed below are defined in order for the communication between the multicast manager 6 and the PIM 8 in the upper layer.

Multicast Manager Data Request: the PIM 8 issues this primitive to the multicast manager 6. The multicast manager 6 transfers the data specified by the primitive.

Multicast Manager Data Indication: the multicast manager 6 issues this primitive to the PIM 8 in order to notify that the manager 6 has received a multicast packet.

Multicast Manager Bind Resource Request: the PIM 8 issues this primitive to the multicast manager 6. The multicast manager 6 binds resources to the port specified by the primitive. With this primitive, the PIM 8 passes the port ID and the multicast channel number which is required if the type of the port specified by the port ID is MULTICAST LISTENER.

Multicast Manager Bind Resource Confirmation: the multicast manager 6 issues this primitive to the PIM 8 in order to return the results of the procedure executed in response to the Multicast Manager Bind Resource Request.

2.4.9 Communicating with NIM

The multicast Manager 6 communicates also with the NIM 9 to carry out the multicast data transfer. The service primitives listed below are defined in order for the communication between the multicast manager 6 and the NIM 9 in the upper layer. The NIM 9 uses these primitives to reset or initialize the status of the multicast manager 6.

Multicast Manager Control Request: the NIM 9 issues this primitive to the multicast manager 6. By this primitive, parameters listed below are passed:

RESET:	Resets the status of the multicast manager 6.

INITIALIZE:	Initializes the status of the multicast manager 6.
TALKER INFO RECEIVED:	Notifies that the NIM 9 has received a talker information reporting packet.

Multicast Manager Control Confirmation: the multicast manager 6 issues this primitive to the NIM 9 in order to return the results of the procedure executed in response to Multicast Manager Confirmation Request.

Multicast Manager Event Indication: the multicast manager 6 issues this primitive to the NIM 9 in order to report the internal status of the multicast manager 6.

2.4.10 Bus reset handling

The operation of the bus reset is described below. If a new node is introduced into the bus, the bus reset is executed to reset all the paths, and then the paths are reconstructed with allocating new node IDs to the respective nodes. In this case, the bus reset complete notification procedure is executed after the bus reset handling procedure. The bus reset handling procedure is illustrated in Figure 22. If the bus reset occurs, the NIM 9 issues a reset request (Multicast Manager Control Request (Reset)) to the multicast manager 6, which then launches the bus reset handling procedure. Upon receiving the request, the multicast manager 6 resets a packet counter to "0" in step S800. Further, the multicast manager 6 clears the multicast channel number stored in the multicast information table, and rejects the multicast transmission request from the PIM 8 in step S810, thereby terminating the bus reset handling procedure.

After the bus reset is completed, the NIM 9 issues Multicast Manager Control Request (Initialize) to the multicast manager 6. Upon receiving the request, the multicast manager 6 executes the bus reset complete notification procedure. The bus reset complete notification procedure is illustrated in Figure 23. If the multicast manager 6 stores a port ID in its own multicast information table and a multicast channel number corresponding to that port ID is cleared, this means that the multicast channel number bound to the port is cleared by the bus reset. Thus, in step S820, the multicast manager 6 extracts a talker entry, whose resource is cleared in the multicast information table. Then, in step S830, the manager 6 broadcasts a talker information reporting packet containing the resource (the multicast channel number). In step S840, it is tested if the resource is newly obtained. If the resource is obtained, the procedure goes forward to step S850, where the resource (the obtained multicast number) is bound to the talker entry. Then, in step S860, the PIM 8 is notified of the result of the resource binding. In this case, the PIM 8 is notified of the successful resource binding. On the other hand, if it is recognized that the resource allocation has failed in step S840, the PIM 8 is notified of the failure. The steps S840 and S850 are illustrated in dotted blocks as the same reason as in the former section 2.4.5 (Binding multicast resources). Then, in step S870, it is tested if there is a remaining talker entry to which no multicast resource is bound. If there is such an entry, the multicast resource binding procedure is carried out again with returning back to step S820 in order to bind the resource to all the talkers. If all the multicast talkers are bound with the multicast resources, the procedure is terminated.

2.5 PIM (Path Information Manager)

The PIM (Path Information Manager) 8 in the mLAN transport layer 10 will be described hereunder. The PIM 8 is provided in the mLAN transport layer 10, and the PIM 8 sets or configures a logical path between a talker port and a listener port. The PIM 8 saves the path information representative of the path already configured, and the PIM 8 automatically reconfigures the path after the initialization caused by a bus reset. However, the PIM 8 maintains the path information only for the ports to handle the two transmission modes, which are the isochronous transmission and the multicast transmission, among the three transmission modes defined in the mLAN transport layer 10.

The PIM 8 provides services as listed below for upper layers.

A. Path configuration/release: Configures a path between ports handling the isochronous transmission and the multicast transmission. Configured paths can be released as well.
B. Path information memory: Stores the path information configured by the PIM 8 services. The PIM 8 stores the path information even while the power is turned off, and the logical paths are reconfigured according to the stored path information.
C. Reconfiguration of path after power-on: After the initialization of the bus caused by power-on, the PIM 8 automatically reconfigures the paths and restores the initial status of the path configuration.
D. Path information exchange: Upon request of host applications, the PIM 8 makes inquiry about the path information to other nodes using a path information administration protocol. The other nodes include a floppy disk drive, for example.

The PIM 8 also provides a service to set and reset a path between a listener port on its local node and a talker port on a remote node. However, the PIM 8 cannot configure a path between talker and listener ports if they are both located in a remote node. This facility is provided by the bus manager as a host application.

2.5.1 Path setting

The path configuration procedure of the PIM 8 is illustrated in Figure 24. The procedure is launched when the PIM 8 receives a request "PIM Set Path Request" from the host application. Upon receiving the request, the PIM 8 obtains the property of a relevant talker port in step S300. Then in step S310, the PIM 8 sets the same property for the relevant listener port as the talker's property, which is previously acquired in step S300. In step S320, the PIM 8 requests the lower layer module (the isochronous manager 7 if the port type is isochronous, or the multicast manager 6 if the port type is multicast) to bind resources. The resource is bound only to the listener in this stage, since the talker is already bound with the resource. If the path is successfully set, the path information table is up-dated in step S330. In this update, a connection flag is changed from "Disconnected" to "Connected". In step S340, the result of path configuration is reported to the host application in response to the request therefrom. In this case, the success of the path configuration is notified. The above-described path configuration never fails, because the actual path configuration is done by copying the already obtained resources of the talker to the listener. Thus, the path configuration is completed.

2.5.2 Data transmission to path

Data transmission to the path is described hereunder referring to Figure 25. The transmission procedure is commenced when the PIM 8 receives data to be transmitted from the upper layer. In step S360, the type of the port to transmit data is identified by referring to the port information entries, which are maintained by the NIM 9 and formatted as shown in Figure 8. In step S370, it is tested whether the port type is talker or not. If the port type is talker, it is tested whether the port type is multicast or not in step S380. If the port is found to be a multicast port, the PIM 8 transfers the data to the multicast manager 6. The multicast manager 6 executes the multicast transmission procedure as described above. If the port is found to be an isochronous port, the procedure goes forward from step S380 to S400, where the data is sent to the isochronous manager 7. The isochronous manager 7 executes the isochronous transmission procedure shown in Figure 12, thereby terminating the transmission procedure. If it is detected that the port is not a talker in step S370, the procedure is terminated.

2.5.3 Releasing path

The path release procedure is shown in Figure 26. The existing path information is stored in the path information table maintained by the PIM 8 on the listener node. The PIM 8 executes the actual path releasing procedure. The path is released without confirmation by the talker. When the PIM receives a request "PIM Release Path Request" from the upper layer, the path release procedure is launched. In step S410, the PIM 8 checks that the path specified by the request is set for the relevant listener port by searching the path information table shown in Figure 4. In step S420, it is tested whether the path is held in a connected state or not. If the path is connected, the resource bound to the listener port is released, and the corresponding entry is deleted from the path information table in step S430. Then, in step S440, a response to the request is returned to the upper layer. In this case, the successful release is reported. If the path is not in the connected state in step S420, the corresponding response is returned to the upper layer in step S440.

2.5.4 Bus resetting, talker information handling, and unsuccessful event notification

The mLAN transport layer 10 dynamically allocates a channel to use the asynchronous multicast transmission, so that the talker must obtain a new multicast channel number upon bus reset. Thus, the multicast channel number bound to a talker may differ before and after the bus reset. Consequently, the listener cannot receive data without setting the same channel number as the talker after the bus reset. Thus, the PIM 8 executes a bus reset handling procedure at first, and then executes a talker information update procedure. The bus reset handling procedure is described below referring to Figure 27. The bus reset handling procedure is launched when the NIM 9 issues a request "PIM Control Request (Reset)" to the PIM 8 upon bus reset. On receiving the request, the PIM 8 sets all connection flags of the entries in the path information table to "DISCONNECTED" in step S450. In this state, the path information table is retained, but the data transfer according to the path information is disabled. After the bus reset is completed, all the nodes re-bind resources to isochronous and multicast talkers in their nodes. If the resource re-binding is successful, the updated information concerning the talkers are broadcasted in the form of the talker information reporting packet formatted as shown in Figure 9.

The arrival of the talker information reporting packet is notified from the multicast manager 6 or the isochronous manager 7 to the PIM 8 (via IM Event Indication (), or MM Event Indication ()), so that the talker information update procedure is launched. The talker information update procedure is shown in Figure 29. In step S460, the PIM 8 in each node searches the node unique ID in the path information table to find out the entry of the talker relevant to the received talker information reporting packet. Then, the property of that talker is identified in step S470. Further, the property of the lis-

For efficiency, the MIDI message can be sent in a packet as it is. Any data, which does not require real time handling, can be transmitted with other protocols such as a normal file transfer protocol (FTP). The data such as digital audio data can preferably be transmitted as a stream by the isochronous transmission method.

Figure 36 shows another embodiment of the inventive mLAN. This embodiment basically has the same construction as that of the previous embodiment shown in Figure 1. Therefore, corresponding blocks are denoted by the same reference numerals as those of the previous embodiment to facilitate understanding of this embodiment. The present mLAN is implemented by a computer system in which all of the nodes #1-#4 are implemented in the form of a personal computer, and are controlled by each CPU. The mLAN system is operated according to application programs loaded into each of the nodes #1-#4 by means of a machine-readable media 50 such as an optical memory disc and a magnetic memory disc.

Namely, as shown in Figure 36, in the mLAN according to the invention, the plurality of the nodes #1-#4 are communicable with one another to exchange information and to transfer data, and the bus is provided for connection to the nodes #1-#4 and for configuration of the logical or virtual paths to logically link one node to another node so as to secure transfer of the data. Each node has at least one port which is allocated for accessing the bus and which is classified into the four types of the isochronous talker, the isochronous listener, the multicast talker and the multicast listener. For example, the node #1 binds an isochronous channel number and a communication band to the isochronous talker 1-T1 when the same is allocated for isochronously transmitting the data labeled by the bound isochronous channel number to the bus through the bound communication band. The node #4 binds an isochronous channel number to the isochronous listener 4-R2 when the same is allocated so that the isochronous listener 4-R2 can exclusively receive data transmitted from another isochronous talker 1-T1 allocated to another node #1 if the transmitted data is labeled by the same isochronous channel number as that bound to the isochronous listener 4-R2. The node #3 binds a multicast channel number to the multicast talker 3-T1 when the same is allocated for asynchronously broadcasting data labeled by the bound multicast channel number to the bus. The node #2 binds a multicast channel number to the multicast listener 2-R1 when the same is allocated so that the multicast listener 2-R1 can exclusively receive data transmitted from another multicast talker 3-T1 allocated to another node #3 if the transmitted data is labeled by the same multicast channel number as that bound to the multicast listener 2-R1.

In a specific form, a transmitter node having either of the isochronous talker and the multicast talker operates when the logical paths are reset for broadcasting talker information representative of resources including an isochronous channel number, a communication band and a multicast channel number, which may be newly bound, respectively, to the isochronous talker and the multicast talker upon resetting of the logical paths, while a receiver node having either of the isochronous listener and the multicast listener operates when the broadcasted talker information is received for newly binding resources including an isochronous channel number and a multicast channel number to the isochronous listener and the multicast listener to thereby restore the logical paths. In such a case, the receiver node saves path information representative of the logical paths and newly binds the resources according to the broadcasted talker information and the saved path information to ensure correspondence between the resources of the receiver node and the resources of the transmitter node in terms of the isochronous channel number and the multicast channel number. Each node has a specific communication architecture composed of protocol layers including a transport layer which contains an isochronous manager for acquiring isochronous resources and binding the acquired isochronous resources to either of the allocated isochronous talker and the isochronous listener, a multicast manager for acquiring multicast resources and binding the acquired multicast resources to either of the allocated multicast talker and the multicast listener, and a path information manager cooperating with the isochronous manager and the multicast manager for providing an upper protocol layer with a service to set the logical paths and to manage the set logical paths. Normally, the transmitter node is allocated with both of the isochronous talker and the multicast talker such that the transmitter node repeatedly carries out a transfer cycle containing isochronous transmission of data by the isochronous talker and asynchronous broadcasting of data by the multicast talker, and such that the transmitter node distributes the data each transfer cycle to either of the isochronous talker and the multicast talker according to property of the data and availability of the isochronous talker and the multicast talker. In such a case, the transmitter node treats mixture of continuous music data and discrete music data, and distributes the continuous music data to the isochronous talker while distributes the discrete music data to the multicast talker.

As described in the foregoing, according to the present invention, virtual and logical paths can be configured within the network independently on the physical connection topology. If two devices are connected to the network, a path can be established between the two devices independently on the physical location thereof, and data can be exchanged through the path. Thus, the data can be exchanged via the virtual paths independently on the network topology, so that the connecting order of the multiple devices is never mistaken or irrelevant, and the data can be transferred reliably. The data link can be modified easily without changing the physical connection of the devices, since the path is configured logically among the devices. The virtual path information is stored in each device connected to the network. Therefore, all the path information of the whole network system can be saved in a memory media, and the same network configuration can be restored easily and instantly with loading the saved path information from the media. The present invention makes it possible to use not only the isochronous transmission method in which the bandwidth of the transmission